# Semantic Content Mediation and Acquisition: The Challenge for Semantic e-Business Solutions

***Jesús Contreras[1], Oscar Corcho[2], V. Richard Benjamins[1]***

[1]Intelligent Software Components, Spain, www.isoco.com, jcontreras@isoco.com
[2]University of Manchester, UK, http://www.cs.man.ac.uk/~ocorcho/, ocorcho@cs.man.ac.uk

## Introduction

A Top Quadrant report[1] situates the Semantic Web within the current Innovation Wave of "Distributed Intelligence". This is one of the main innovation waves of the last centuries including textile, railway, auto, computer, distributed intelligence (1997-2061) and nanotechnology (2007-2081). The Distributed Intelligence wave started in the late nineties and is expected to peak between 2010 and 2020. The report estimates first return on investments in 2006-7, growing to a market of $40-60 billion in 2010. Funds are coming primary from governments, venture capitalists and industry commercialization. Over the next few years, this is expected to change in favour of industry commercialization. Figure 1 illustrates this graphically.

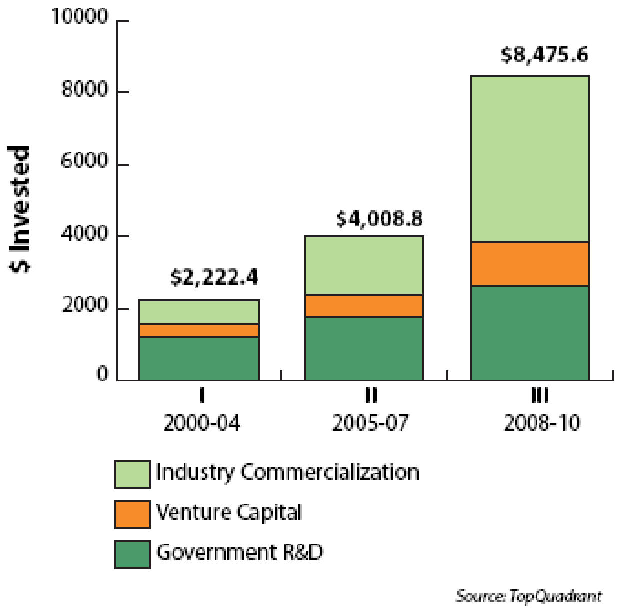



**Figure 1** Semantic Technology investment to 2010 by type ($US Billions)

Of course, for those predictions to become true, mature, reliable and scalable technology is required. Some of this technology is already in place and currently exploited in the market by early adopters. Other technology is still under development. There are at least two main application areas of Semantic Web technology: Knowledge Management and Enterprise Application Integration, respectively within and between enterprises.

For knowledge management, the main current challenge is to semantically annotate existing content (information) such that it becomes machine-readable. Many organizations are suffering from the Information Overload problem and their knowledge workers spend a significant amount of productive time to searching for the right information. Semantic Web technologies offer a solution to this in the form of semantic search engines that, instead of performing

[1] EVALUATION AND MARKET REPORT. A report on assessment of the potential market, including guidance as to directions for further development in response to evolving industrial requirements. http://wonderweb.semanticweb.org/deliverables/documents/D25.pdf

keyword-based search, do ontology-based search. In this chapter, we will discuss and present some of the tools and techniques available for adding semantic annotations to legacy content.

For Enterprise Application Integration (EAI) and Business to Business Integration (B2Bi), one of the main challenges lies in solving the interoperability problem. Basically, the problem here is to integrate (make work together) systems that have been designed to work standalone. The main problem resides here in building bridges relating the different systems. We will see that there are different types of such “bridges” or so-called “mediators”: bridges to mediate between different data models, services/protocols and business models. There is work underway to construct technology that supports the mediation process, either at design time or at run time.

Organizations interested and committed to attack the problem of Information Overload or EAI/B2Bi have to take a decision whether to use existing technology or Semantic Web technology. Examples of questions that such organizations face include (but are not limited to):

- Standardization. Is Semantic Web technology compatible with existing industry standards already used in the organization?
- How do I calculate the costs and benefits of deploying Semantic Web technology in my business? What is the total cost of ownership compared to more traditional technologies

The rest of this chapter explains in detail the technologies available for the two main problems identified: how to mediate between heterogeneous content and how to add semantics to existing content.

## Semantic Content Mediation

This chapter focuses on the problems and solution directions for semantic content mediation at the three different levels identified in the introduction: data, service/protocol and business process.

There are several theories about how the Semantic Web standards take shape in each sector. The idea on having one single standard is non real, unless there is a clear sector monopoly (e.g.: governmental sector, defense, etc.). The usual scenario is well described in the [Forrester Internet X report 2001] describing how sector leaders establish several standards de facto and how suppliers and satellite companies adopt them.

The expected scenario considers having several standards in each sector that are easilly interoperable. In this sense, there are several research and technological initiatives that aim at providing methods for industrial Semantic Interoperability at low cost and in a reliable way.

This chapter introduces the high-level conceptual model for mediation, as this particular area is called, at these three levels:

- Data mediation is concerned with how to transform the messages exchanged by different applications. In the context of service-oriented architectures, these messages are normally exchanged by Web services in the form of SOAP messages. The transformation to be done includes not only the syntax of the messages, including their internal structure, but in many situations also their semantics.
- Service/Protocol/Process mediation is concerned with the different interaction patterns of the applications participating in a conversation. This different interaction patterns may be due to the fact that the processes followed in the two sides of the conversation are not completely equal (one of them may require checking the validity of a credit card before performing the payment, as two separate processes, while the other may perform the validity check and the payment at the same time).
- Business process mediation is concerned with a more high-level type of application alignment, which is not related to the types of messages nor to their interaction patterns, but with the conceptual models used for carrying out the processes involved in each side of the

conversation (in a selling process one of the entities follows a closed auction model while the another uses a continuous auction model).

### Data Mediation

As specified in the introduction, there are several aspects to be considered in the context of data mediation, regarding the syntax and structure of the messages exchanged between the applications involved in a communication, and the semantics of the content exchanged.

Let us see first some of the issues that arise from the use of **different message syntax and structure**.

Early distributed applications (mainly in client-server architectures), were built using Remote Procedure Calls (RPC), which enabled communication between processes on different machines. RPCs were based on extending the notion of conventional, or local procedure calling, so that the called procedure need not exist in the same address space as the calling procedure. The two processes may be on the same system, or they may be on different systems with a network connecting them. General purpose programming languages like C, C++ or Java, and distributed middleware like DCOM and CORBA, among others, give support to this paradigm.

However, RPCs represented a compatibility and security problem when they are used in open distributed environments where the communicating applications are running on different platforms (including operating systems, technologies and programming languages) and are not inside the same organisation. To alleviate this problem (especially that of compatibility), the W3C has proposed the use of SOAP[2] (Simple Object Access Protocol), an XML-based protocol that supports HTTP-based communication between applications by means of Web services.

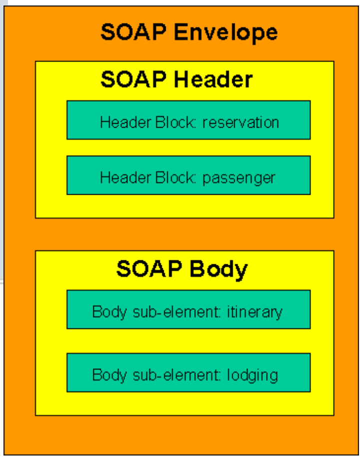
SOAP (Simple Object Access Protocol) defines a simple messaging framework for transferring information specified in the form of an XML infoset between an initial sender and an ultimate receiver. Many of the communications may involve the participation of other processing nodes - called SOAP intermediaries.

A SOAP message can be considered like an envelope that contains two parts, the header and the body, from which the first one is optional. The SOAP body is the element that contains the main end-to-end information. Its contents are implicitly targeted and are expected to be understood by the ultimate receiver. The SOAP header is an extension mechanism that provides a way to pass information that is not application payload, that is, control information like passing directives, contextual information related to the processing of the message, etc. This allows a SOAP message to be extended in an application-specific manner.

SOAP also provides a model for handling situations when faults arise in the processing of a message.

Figure 2 provides an example of a message that requests a return flight between New York and Los Angeles to a travel reservation service.

```
<?xml version='1.0' ?>
<env:Envelope xmlns:env="http://www.w3.org/2003/05/soap-envelope">
 <env:Header>
  <m:reservation xmlns:m="http://travelcompany.example.org/reservation"
          env:role="http://www.w3.org/2003/05/soap-envelope/role/next"
           env:mustUnderstand="true">
   <m:reference>uuid:093a2da1-q345-739r-ba5d-
pqff98fe8j7d</m:reference>
   <m:dateAndTime>2001-11-29T13:20:00.000-05:00</m:dateAndTime>
  </m:reservation>
  <n:passenger xmlns:n="http://mycompany.example.com/employees"
          env:role="http://www.w3.org/2003/05/soap-envelope/role/next"
           env:mustUnderstand="true">
   <n:name>Åke Jógvan Øyvind</n:name>
  </n:passenger>
 </env:Header>
 <env:Body>
  <p:itinerary
    xmlns:p="http://travelcompany.example.org/reservation/travel">
```




[2] http://www.w3.org/TR/soap/

```
  <p:departure>
    <p:departing>New York</p:departing>
    <p:arriving>Los Angeles</p:arriving>
    <p:departureDate>2001-12-14</p:departureDate>
    <p:departureTime>late afternoon</p:departureTime>
    <p:seatPreference>aisle</p:seatPreference>
  </p:departure>
  <p:return>
    <p:departing>Los Angeles</p:departing>
    <p:arriving>New York</p:arriving>
    <p:departureDate>2001-12-20</p:departureDate>
    <p:departureTime>mid-morning</p:departureTime>
    <p:seatPreference/>
  </p:return>
 </p:itinerary>
 <q:lodging
  xmlns:q="http://travelcompany.example.org/reservation/hotels">
  <q:preference>none</q:preference>
 </q:lodging>
</env:Body>
</env:Envelope>
```

**Figure 2.** SOAP envelopes as a means of communication (adapted from [SOAP_Primer]).

The use of SOAP solves most of the compatibility problems in the communication between processes that are running in different organisations, with different operating systems and using different technologies and programming languages. However, the initial sender of a SOAP envelope must know exactly what is the structure of the content that the ultimate receiver of the message expects in order to allow for an effective communication. This includes the number and type of parameters that the receiver uses, the order in which they are sent inside the message body, the format used for datatypes, etc.

This problem is being tackled by Semantic Web initiatives for (Semantic) Web service description and execution, such as OWL-S[3] and WSMO[4] (Web Service Modelling Ontology). These initiatives propose the use of Semantic Web languages like RDF or WSML, respectively, for expressing the content of the messages sent inside SOAP envelopes. The use of these languages allows for more flexibility in the models used for exchanging data: the order in which the parameters are sent is not important, the datatype formats are standardised through the use of XML Schema datatypes, and the parameters are clearly characterised according to RDF Schema, OWL or WSML ontologies, as we will see later.

The second problem that we outlined at the beginning of this section (**different semantics of the content exchanged**) is the major challenge in data mediation. The applications involved in a conversation may be using different conceptual models to represent the data that they are exchanging. While the sender may use the parameter departurePlace as the airport name, the receiver of the message may be expecting that the departurePlace is the name of the city and the name of the country from which the travel will depart.

Above we have already talked about the use of ontologies for the representation of the data that is sent around inside SOAP messages. Ontologies are shared models of a domain, what means that if the two applications involved in a conversation share the same ontology then there will be no problems on understanding the content of the messages that they are exchanging.

However, this is not always the case when any two applications are exchanging messages. That is, there are cases where the message sender and receiver do not share the same ontology. In that case, there will be differences in the conception of the domain that may affect the effective communication between them. These differences are commonly known as mismatches.

Ontology alignment methods, techniques and tools are applied to solve these mismatches. They usually consist in establishing mappings between the concepts and relations of the ontologies used by the applications (usually known as mapping rules, even though they are not necessarily

[3] http://www.daml.org/services/owl-s/
[4] http://www.wsmo.org/

implemented as rules), and execute those mappings to transform the messages that are being exchanged. Figure 3 illustrates this process, showing how the message content would be transformed according to the defined mapping rules.

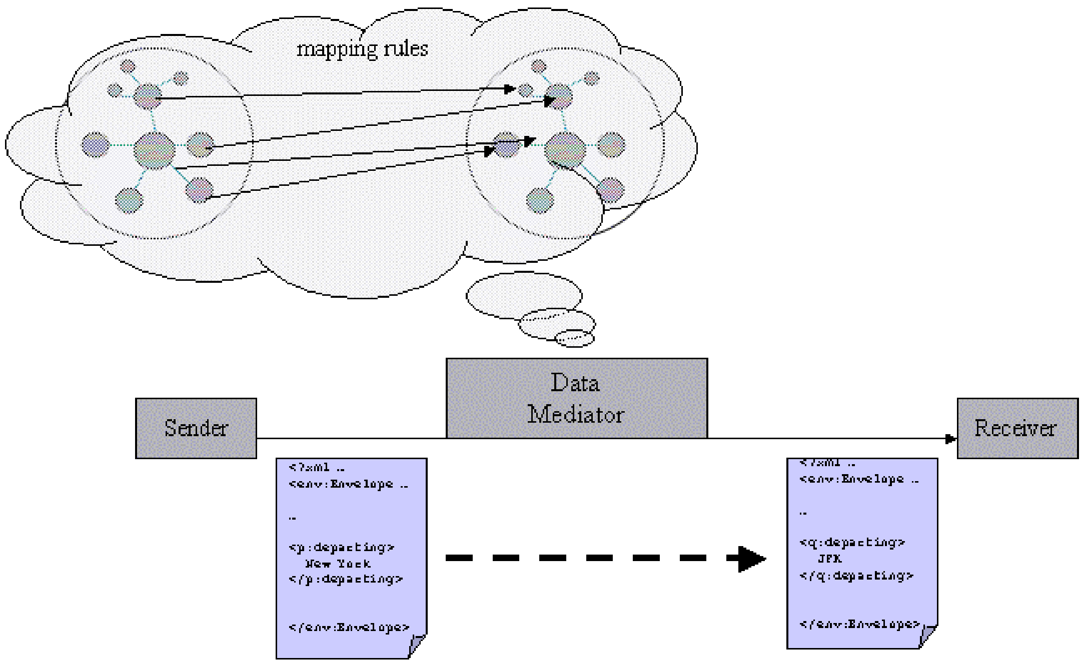



**Figure 3.** Ontology mediation for the exchange of messages between Web services.

For a deeper description of ontology mediation, we refer to [Euzenat et al., 2004]. They provide a good overview and classification of the state of the art on ontology alignment methods and tools, as well as a characterisation of the different types of ontology mismatches that can be found in applications and a set of use cases for ontology mediation, which include its use in Web services. In [Mocan and Cimpian, 2004] there is also a description about the data mediation component in the context of the WSMO initiative.

### *Service/Protocol/Abstract Process Mediation*

The parties involved in a conversation may have not only the needs for data transformation described in the previous section, but also different message interaction models. It is at this stage where service mediation (aka protocol mediation or abstract process mediation) is needed.

Let us see first how the message interaction patterns to be used in a conversation are expressed in the context of service-oriented architectures. In the Web services world, two different types of interactions are distinguished: orchestration and choreography. *Web service orchestration* describes how Web services can interact with each other at the message level, including the business logic and execution order of the interactions. *Web service choreography* tracks the sequence of messages that may involve multiple parties and multiple sources (requesters, providers, etc.). With the last enhancements and standards this distinction has almost disappeared and in many cases both aspects are generally called as Web service orchestration.

Early work in Web services orchestration included eCo (from CommerceNet, focused on the integration of e-commerce services), WSCL (Web Services Conversation Language, from HP), XLANG (from Microsoft, included in the Microsoft Biztalk Server) and WSFL (Web Services Flow Language, from IBM).

Some of these early proposals were superseded by other languages, and currently the two most important proposals are the following:

- BPEL4WS [BPEL4WS] (Business Process Execution Language for Web Services). It superseded XLANG and WSFL and now is being further developed with a different name, WS-BPEL [WS-BPEL], under the standardisation process in OASIS.
- WS-CDL (Web Services Choreography Description Language). It evolves from WSCI [WSCI] (Web Service Choreography Interface) and BPML [BPML] (Business Process Management Language), which in their turn evolved from WSCL. This language is now following the standardisation process in the W3C.

All these languages and proposals allow establishing the order in which a set of Web services have to exchange their messages in order to create a more complex business process. In other words, they allow specifying the control flow of composite business processes by means of allowing an effective communication between the Web services involved in the process, including sequential and parallel activities, loops, etc. They provide execution engines that are able to read such specifications and drive the communication, and they also take into account aspects such as the transactionality of operations, callback mechanisms, etc.

By allowing the expression of the control flow that a set of Web services must abide to in order to form a complex service, these languages aim at reducing the complexity required to orchestrate Web services, thereby reducing time-to-market and costs, and increasing the overall efficiency for building a complex service. However, the interaction patterns of several Web services involved in a communication do not always have a precise match, that is, do not always follow exactly the same pattern in realising the complex process. Sometimes there can be mismatches like the ones identified in [Cimpian and Mocan, 2005], and summarised in Figure 4:

- *Unexpected messages*. One of the parties receives does not expect to receive a message issued by another. For instance, in a commercial transaction with a credit card a service sends the credit card type, the credit card number, the expiration date, the full name and the pin code, while the service that receives those messages does not expect to receive a pin code, since it does not use it.
- *Messages in different order*. The parties involved in a communication send and receive messages in different orders. In the previous case the sender may send the messages in the order specified above while the receiver expects first the full name and then the rest of the messages.
- *Messages that need to be split*. One of the parties sends a message with multiple information inside it, which needs to be received separately by the party with which it is communicating. In the previous example, the sender sends the expiration date in one message, while the receiver expects it as two messages, indicating the expiration month and the expiration year, respectively.
- *Messages that need to be combined*. One of the parties sends a set of messages that the receiver expects as a single message with the multiple information. In the previous example we can think of the inverse situation to the one presented (the expiration date is sent in two different messages and the receiver expects it in a unique message).
- *Dummy acknowledgements or virtual messages that have to be sent*. One of the parties expects an acknowledgement for a certain message, but the receiver does not issue such acknowledgement; or the receiver expects a message that the sender is not prepared to send.

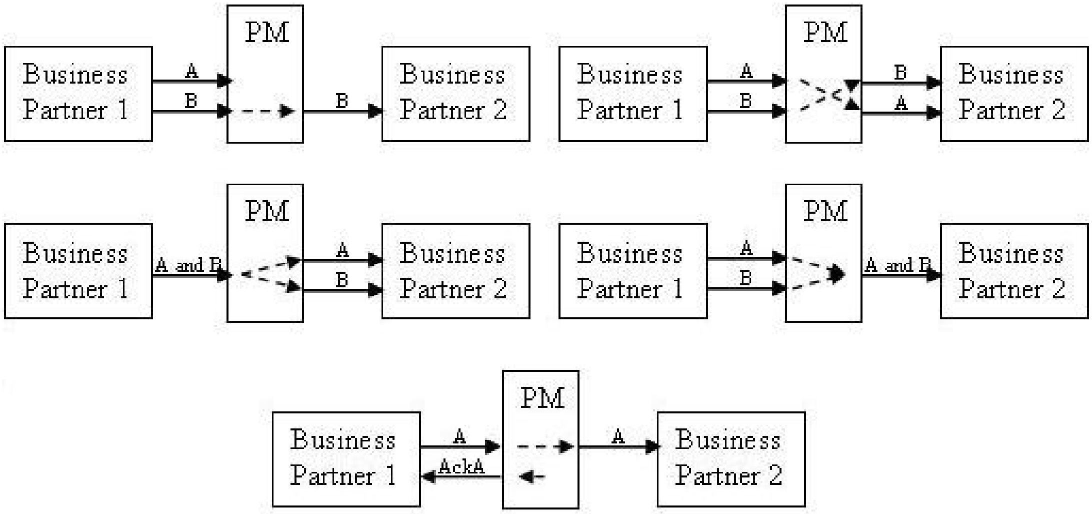



**Figure 4.** Set of message interaction mismatches that can appear in a Web service communication (from [Cimpian and Mocan, 2005]).

All these mismatches can be combined to form other types of mismatches, hence adding more complexity to the task of dealing with mismatches. In fact, after the definition of a complex service one must determine the compatibility of the external visible behaviours of the Web Services involved in the communication, and determine the correctness and validity of the resulting complex service.

The previous languages are not built specifically to deal with these types of mismatches. Consequently, there is the need of building a new component for these architectures, which is usually called service, protocol or abstract process mediator. This is a current research effort in the context of Semantic Web Services.

WSMF [Fensel et al, 2002] (Web Service Modelling Framework) goes further than the previous approaches in the aim of reducing the complexity required to orchestrate Web services. It aims at achieving automation in the process of composition of Web services to create more complex ones. This framework introduces the idea of using strong mediation mechanisms to enable an effective communication by means of using semantics to describe Web services.

Based on this framework [Preist et al., 2005] describe the implementation and use of a solution in the logistics domain. This solution is based on the existence of a general abstract state machine that represents the overall state of the communication between parties, and a set of abstract machines for each of the parties in the conversation, which specify their state and the sets of actions to be performed when they receive a set of messages or when they have to send a set of messages.

Furthermore, in the context of the WSMO initiative, [Cimpian and Mocan, 2005] describe the approach taken for the design and implementation of the process mediator for the Semantic Web Service execution engine WSMX. This approach is similar to the one described in [Preist et al., 2005], since it is based on the use of an abstract machine with guarded transitions that are fired by the exchange of messages and the definition of choreographies for each of the parties involved in the communication.

### *Business Process Mediation*

Sometimes the applications involved in a conversation do not only differ in the message syntax and structure, on the message semantics and on the sequencing of the messages being exchanged, but also on their business models. The example shown in the introduction, where two applications based on different auction models have to interact, is a good example of how two interacting applications may differ at the business level.

Unfortunately, due to the high diversity of applications and application domains where this kind of mediation has to be performed, the solutions proposed for solving the business process mediation problem are usually based on ad-hoc solutions, hence with a high development cost.

# Provision of Semantic Content and Applications

## *Introduction*

The emerging paradigm of the Semantic Web offers a great opportunity for advanced application development based on a common, formal and shared formalism, called ontologies. The existence of domain ontologies gives us an opportunity to construct highly reusable knowledge bases. The well known problem of knowledge acquisition, or ontology feeding in this case, becomes a worthwhile challenge in the proliferation of these kinds of technology and applications.

On the one hand the introduction of Semantic Web technologies into data description allow for revolutionary application of information delivery (search engines, personalized data presentations, advertisement, etc.) as well as for advanced applications with reasoning capabilities.

On the other hand the usage of semantic standards in service descriptions allows reducing the traditionally long distance between the high level business process definitions and their implementations using IT technologies. The road form the conception of a new service or product designed at management level to its delivery by the IT systems can be reduced using semantic description of primitive services. The so called "Semantic Service Oriented Architecture" allows for building of new services by semiautomatic combination and cooperation of existing services.

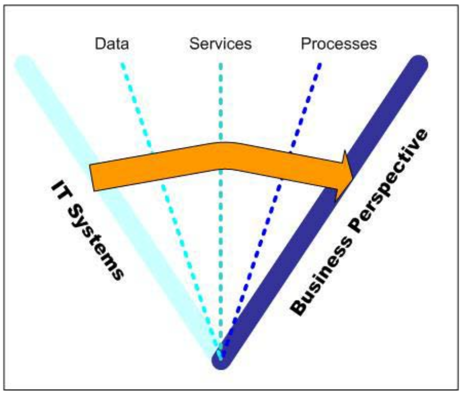



**Figure 5:** The road from IT Systems into Business Process Perspective

The potential bottleneck may arise when trying to introduce real data into semantic enabled solutions and applications. The effort and cost spend on the translation of existing non-semantic data into semantic standard depends on the application domain and the original information format. In case of application working already on structured data (billing, e-procurement, etc.), based on standards, the step into the Semantic Web technology might become smooth. In case of knowledge intensive applications (report analysis, document search and management, CRM, etc.) the effort on obtaining precise semantic data from unstructured sources might constitute a serious barrier. For the domain definition some of the parameters taken into account are: the application scope (e.g.: timeline in the model, general purpose upper model, alignment with standards), the information granularity (how much detail attributes, relations needed in the model, etc), the information quality (need for frequent updates, consistency checking, intelligent integration, etc.). All these parameters are determined in the application analysis phase when designing and constructing the domain ontology.

The other dimension taken into account is the information provenance and its original format. For structured databases, in principle, the process of adding semantics to the information does not necessarily imply large effort. If there is no need for heterogeneous sources integration and the application scope is similar to database schema definition, there exist software components [D2R] that allow the data exportation in few mouse clicks.

For unstructured sources, such as free text, non textual sources (pictures, graphs, multimedia) the effort needed for its conversion into semantic formats may become extremely high. The experience in the HALO project shows the cost of modelling one single book page (chemistry text book for secondary schools) into a semantic model that would allow for passing a examinations is about 10.000 U.S. Dollars when doing it manually [HALO].

## *Background*

The provision of a formal and explicit data is not a new problem. In the AI history, first with knowledge-based systems and then with intelligent agents, it was identified as a potential bottleneck [Feigenbaum, E., & McCorduck, P. 1983] for the proliferation of these kinds of applications. There are many initiatives and tools that provide (semi)-automatic support for detection of concepts, properties and relations using text statistics and natural language processing [KAON], [GATE], [Amilcare].

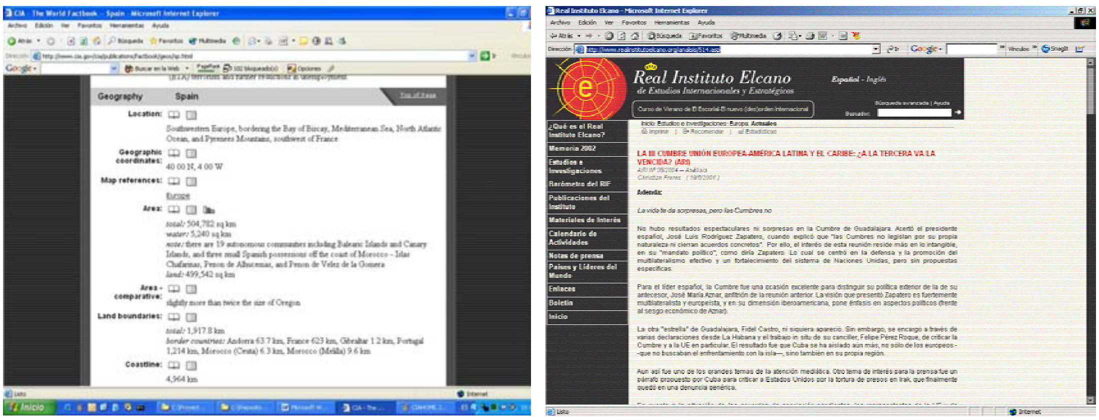



**Figure 6:** Examples of structured (left hand side) and non-structured (right hand side) sources.

In the information extraction field, there are several existing technology approaches. For instance, regular expression matching is very effective in well defined and small domains with highly structured sources. For open domains with structured documents it is worthwhile to render the document extracting its visual structure and process the data semantic from the special order between elements. For documents with weak visual or encoding structure there is a need to use natural language processing techniques that would identify possible instances and relations between them. Also, since some sources might be defined in HTML there could be a need to understand the HTTP protocol and HTML object included in the documents, such as links, forms, text inputs, etc. The term “structured” concerns not only the source structure in terms of HTML tags, but also structure in the final layout rendering of the document or the existence of linguistic syntactic or terminological structures. The Figure 6 shows HTML sources, one with visual structure for its data.

### *Ontology Annotation Tools*

We can define an ontology related Annotation Tool as a tool which makes use of a pre-existing ontology to insert mark-up into a Web page or other document and/or is used to populate knowledge bases via the mediation of a marked up document.

There are thus two main philosophies behind ontology annotation tools. Firstly, what we can call the semantic web approach sees the production of annotated pages as primary with the population of ontologies as secondary. Secondly, what we can call the knowledge engineering approach sees ontology-guided annotation of documents as a means of populating knowledge bases as well as producing annotated documents. In essence both philosophies produce similar tools. However the difference in emphasis is important since it is likely that depending on the goal of the user one or other class of tool will be more appropriate.

In the table below [D3.1 Esperonto Deliverable, updated] we present some of the existing tools that face the problem of obtaining semantically structured data from unstructured sources.

| Annotation Tools | Input Source | Output | Technology | Domain/ Models | Adaptive | Ontology |
|---|---|---|---|---|---|---|
| Citeseer | PDF, PS | Database | Regular Expressions | Scientific publications | No | Implicit |
| Amilcare | Plain or markup | DAML or similar | NLP, ML | Any | Yes | Implicit |
| CREAM | Plain or markup | Any | Mapping rules | Any | No | Explicit |
| AIDAS | PDF | Internal | Text processing | User manuals | No | Explicit |
| Ariandne | HTML | Database | Mediators, Mappings | Any but defined | No | Explicit |
| Google News | HTML | Internal | Unknown | News | Unknown | Implicit |
| Web Scraper | HTML | RDF | Unknown | Any | No | Implicit |
| Getsee | HTML | Database | Regular Expressions | Financial | No | Implicit |
| Whizbang! | HTML | Database | ML and Text processing | Any but defined | Yes | Implicit |
| Knowledge Parser | HTML | Ontologies | NLP, Regular Expressions, Rendering, HTML | Any but defined | No | Explicit |

Studying existing approaches we may conclude that the most successful systems in term of precisions are those which focus on particular and well defined domains with a lot of previously structured sources. The challenge is to construct a generic annotation system that is able to operate on any domain with no large additional effort.

### *The Roadmap to Sector Specific Semantic Standards and Content*

As expressed in some strategic forecasts about the proliferation of the Semantic Web [Forrester Internet X report 2001] there will be established a few ontologies per each business sector. Based on those established and on other standard ontologies, it will possible to build (semi) automatic tools to produce semantically annotated content for data, services and process.

We can observe several sectors that nowadays are developing or already use ontologies (or equivalent semantic formalisms) in their information systems. There are several ways of introducing Semantic Web technology in a particular business for obtaining an acceptable return on investment.

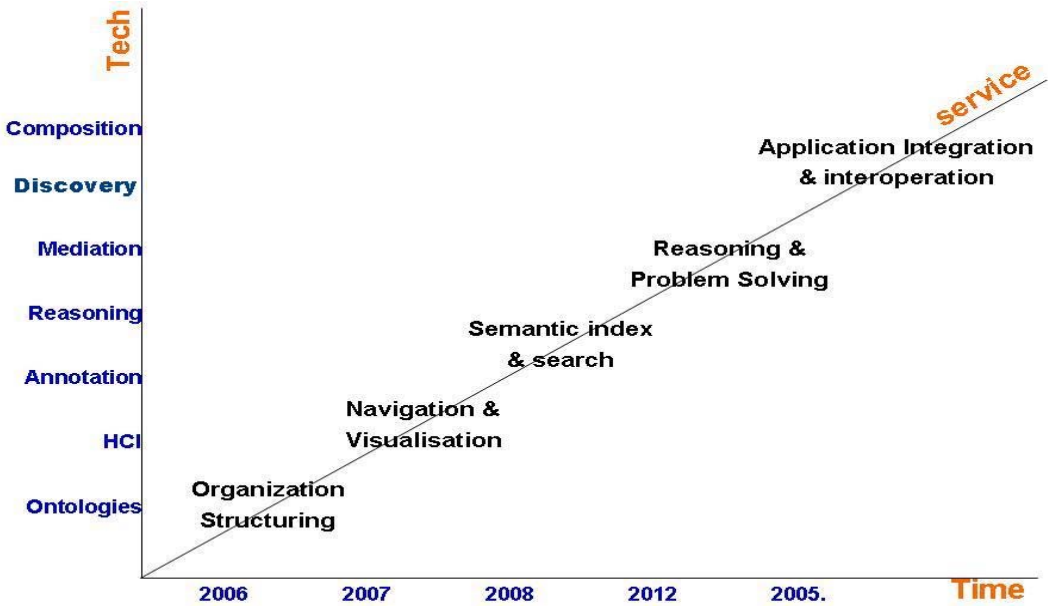



**Figure 7**: Possible SW Application Evolution

Some domains are especially suitable for initial SW technology usage. We point one of them as example of real applications of this technology.

### *Business Case Study: Semantic Portal of International Affairs*

The Royal Institute Elcano (Real Instituto Elcano: www.realinstitutoelcano.org ) in Spain is a prestigious independent political institute whose mission is to comment on the political situation in the world focusing on its relation to Spain. As part of its dissemination strategy it operates a public website. The online content can be accessed by navigating through categories or by a keyword-based, full text search engine. We have used the Knowledge Parser software for feeding a semantic portal, where access is based on the meaning of concepts and relations of the International Affairs domain. In addition to an automatic ontology-based annotator, the portal includes a semantic search engine with a natural language interface, a web publication tool allowing semantic navigation, and a 3D visualization component. The semantic portal is currently being tested by the Institute.

Based on interviews with experts of the Elcano Institute, and using the CIA word factbook (www.cia.gov/cia/publications/factbook/) as the basis for the ontology of International Affairs. The CIA fact book is a large online repository with actual information on most countries of the world, along with relevant information in the fields of geography, politics, society, economics, etc.

We have used the competency questions approach [Uschold, Gruniger 96] to determine the scope and granularity of the domain ontology. On an iterative process knowledge engineers expanded the basic ontology into the final product including information from other four sources. Some examples of competency questions that we considered include:

- `What countries are participating on Iraq campaign?`
- `Who is the head of the state of France?`

- What government type has Georgia?
- How big is the population of Iceland?
- Which are all European Union member countries?
- Which are all agreements between Spain and Brazil subscribed during Da Silva’s govern?

The ontology consists of several top level classes, some of which are:

- Place: Concept representing geographical places such as countries, cities, buildings, etc.
- Agent: Concept taken form WordNet [WordNet 98] representing entities that can execute actions modifying the domain (e.g.: Persons, Organizations, etc.)
- Events: Time expressions and events
- Relations: Common class for any kind of relations between concepts.

The ontology has been constructed in Protégé (http://protege.stanford.edu)

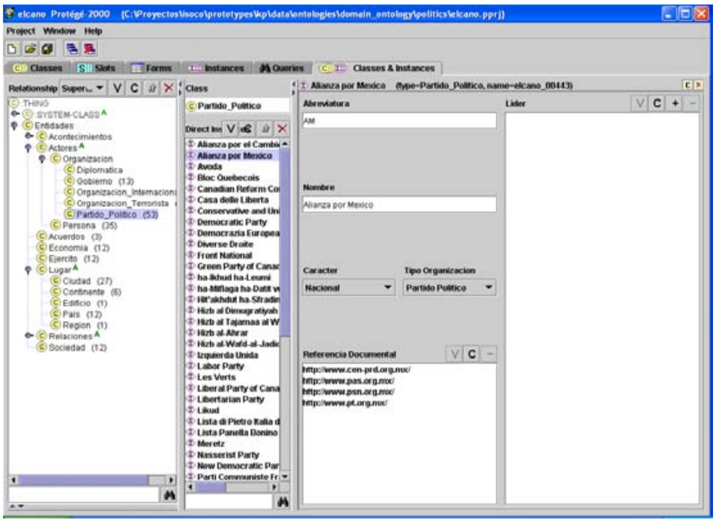


**Figure 8:** Ontology for International Affairs edited in Protégé 2000 tool

The annotation tool, Knowledge Parser, in this version of the application deployed, processes five sources. The first two sources provide information about geopolitical facts of the world countries. Information about populations, economy, border relations, and social indicators are described. These two sources are highly structured from the layout point of view. The KP strategy relies mostly on HTML object interpretation (HTML combo boxes for each country page selection, HTML link for accessing detailed information, etc.) and on layout interpretation for tabular structure processing (label ‘area’ is IN ROW with the area value, section on social information in BELOW the economic section, etc.).

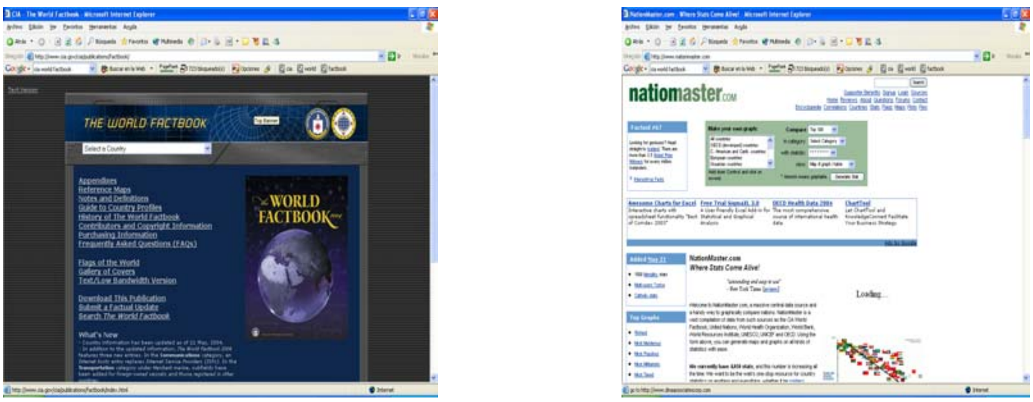



**Figure 9**: CIA Word Fact book and Nation Master Pages

The third, CIDOB, source provides additional information about politicians, governments, political parties, government departments. This information significantly extends the instance set of the domain ontology with new information about persons, organization and their roles in relation with the existing data. There were several challenges when consolidating information for these two new sources: the first one deal with the fact the CIDOB source is written in Spanish. We have used external translation services for translating common words. Some specific terms, almost all proper names, were translated using ad-hoc thesauri built up in collaboration with domain experts.

This application allows for search and browse functionalities thanks to the automatic annotation system of source data.

## Conclusions

In this chapter, we discuss two main applications areas of Semantic Web Technology. On the one hand Enterprise Application Integration and B2B-integration require semantic technologies to overcome interoperability problems to be solved in a scalable manner. On the other hand, automatic annotation of existing content with semantic metadata helps to overcome information load. In each of the areas, we briefly presented both theoretical and practical aspects.

## Acknowledgements

Part of the work described in this project has been supported by the Esperonto and SWWS project, both funded by the European Commission under the FP5-IST framework.

## References

[D2R] Bizer, Christian: D2R MAP – A Database to RDF Mapping Language, WWW2003, The Twelfth International World Wide Web Conference, Budapest, HUNGARY, 2003

[HALO] http://www.projecthalo.com/content/docs/halopilot_vulcan_finalreport.pdf

[Feigenbaum, E., & McCorduck, P. 1983] ] Feigenbaum, E., & McCorduck, P. (1983). *The Fifth Generation: Artificial Intelligence and Japan's Computer Challenge to the World*. Addison-Wesley Publishing Co., Reading, MA.

[KAON] E. Bozsak, Marc Ehrig, Siegfried Handschuh, Andreas Hotho, Alexander Maedche, Boris Motik, Daniel Oberle, Christoph Schmitz, Steffen Staab, Ljiljana Stojanovic, Nenad Stojanovic, Rudi Studer, Gerd Stumme, York Sure, Julien Tane, Raphael Volz, Valentin Zacharias "KAON - Towards a large scale Semantic Web " In Kurt Bauknecht and A. Min Tjoa and Gerald Quirchmayr, E-Commerce and Web Technologies, Third International Conference, EC-Web 2002, Aix-en-Provence, France, September 2-6, 2002, Proceedings, volume 2455 of Lecture Notes in Computer Science, pp. 304-313. Springer, 2002.

[GATE] H. Cunningham. GATE, a General Architecture for Text Engineering. Computers and the Humanities, 36:223-254, 2002

[Amilcare] Ciravegna. Adaptive information extraction from text by rule induction and generalization. In 17th International Joint Conference on Artifical Intelligence, 2001.

[BPEL4WS] Business Process Execution Language for Web Services. http://www-128.ibm.com/developerworks/library/specification/ws-bpel/

[Cimpian and Mocan, 2005] Cimpian E, Mocan A. Process Mediation in WSMX. WSMX working draft D13.7. http://www.wsmo.org/TR/d13/d13.7/v0.1/

[Euzenat et al., 2004] Euzenat J et al. (2004). State of the art on ontology alignment. KnowledgeWeb deliverable D2.2.3. Available at http://knowledgeweb.semanticweb.org/

[Fensel et al., 2002] WSMF: D. Fensel, C. Bussler, Y. Ding, B. Omelayenko: The Web Service Modeling Framework WSMF. Electronic Commerce Research and Applications, 1(2), 2002.

[Mocan and Cimpian, 2004] Mocan A, Cimpian E. Mediation. WSMO working draft D13.3. http://www.wsmo.org/2004/d13/d13.3/v0.1/
[Preist et al., 2005] Chris Preist, Javier Esplugas Cuadrado, Steve Battle, Stuart Williams, Stephan Grimm. Automated Business-to-Business Integration of a Logistics Supply Chain using Semantic Web Services Technology. In: Proceedings of the 4$^{th}$ International Semantic Web Conference (ISWC2005), Galway, Ireland.
[WSCI] WSCI: Web Service Choreography Interface. http://www.w3.org/TR/wsci/
[WS-BPEL] OASIS Web Services Business Process Execution Language (WSBPEL) Technical Committee. http://www.oasis-open.org/committees/tc_home.php?wg_abbrev=wsbpel
[BPML] Business Process Modelling Language Specification 1.0. http://www.bpmi.org/bpml-spec.htm
[D3.1 Esperonto Deliverable] Contreras J., Benjamins V.R., Martin F., Navarrete B.; Aguado de Cea G, Alvarez de Mon I.; Pareja Lora A., Plaza Arteche R: "Annotation Tools and Services" Esperonto Services
[Forrester Internet X report 2001] http://www.forrester.com/ER/Research/Report/Summary/0,1338,11282,00.html
[Uschold, Gruniger 96] M. Uschold and M. Gruninger. Ontologies: principles, methods, and applications. Knowledge Engineering Review, 11(2):93–155, 1996.
[WordNet 98] WordNet An Electronic Lexical Database Edited by Christiane Fellbaum; May 1998 ISBN 0-262-06197-X, MIT Press